Europäisches Patentamt
European Patent Office
Office européen des brevets

REC'D 20 AUG 2004
WIPO PCT

Bescheinigung Certificate Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. Patent application No. Demande de brevet n°

03291705.6

PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN COMPLIANCE WITH RULE 17.1(a) OR (b)

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

R C van Dijk

Europäisches Patentamt | European Patent Office | Office européen des brevets

Anmeldung Nr:
Application no.: 03291705.6
Demande no:

Anmeldetag:
Date of filing: 09.07.03
Date de dépôt:

Anmelder/Applicant(s)/Demandeur(s):

Deutsche Thomson-Brandt GmbH
Hermann Schweer Strasse 3
78048 Villingen-Schwenningen
ALLEMAGNE
Thomson Licensing S.A.
46, quai Alphonse Le Gallo
92648 Boulogne Cedex
FRANCE

Bezeichnung der Erfindung/Title of the invention/Titre de l'invention:
(Falls die Bezeichnung der Erfindung nicht angegeben ist, siehe Beschreibung.
If no title is shown please refer to the description.
Si aucun titre n'est indiqué se referer à la description.)

Error correction method for reed-solomon product code

In Anspruch genommene Priorität(en) / Priority(ies) claimed /Priorité(s) revendiquée(s)
Staat/Tag/Aktenzeichen/State/Date/File no./Pays/Date/Numéro de dépôt:

Internationale Patentklassifikation/International Patent Classification/ Classification internationale des brevets:

H03M13/00

Am Anmeldetag benannte Vertragstaaten/Contracting states designated at date of filing/Etats contractants désignées lors du dépôt:

AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HU IE IT LU MC NL PT RO SE SI SK TR LI

Error correction method for Reed-Solomon product code

The present invention relates to soft-decision decoding of Reed-Solomon product codes. It further relates to a system for correcting errors in a Reed-Solomon product code.

For optical recording media such as digital versatile disks (DVD) a Reed-Solomon (RS) product code is used for error correction. The data frames include horizontal and vertical parity data for correcting errors in the frame per rows and columns. The data rows and columns of the frame constitute so-called product-codes. The outer code is a RS code(208,192,17), while the inner code is a RS code(182,172,11). Therefore, one data frame consists of 208*182=37856 bytes.

Multipass devices save at least two data frames in a memory. One data frame is saved for input/output and the second is saved for correction. Current systems save the data either in external SDRAM or in internal SRAM. Due to random accesses during correction of the errors, the SDRAM approach is significantly slower than the approach using internal SRAM.

US 6,032,283 discloses an implementation of a DVD controller. The input/output streaming and correction is done in an external SDRAM. For fast processing an internal SRAM is used. The syndromes are saved in the internal SRAM, while correction is done in the external SDRAM. During correction the orthogonal syndromes are updated with the error values. Therefore, additional hardware is required, but the process is accelerated. The disclosed implementation has a disadvantage that random accesses to the SDRAM are required, which slow down the correction. The internal SRAM consumption for multipass correction in that implementation is about 2*4992=9984 bytes.

It is an object of the invention to propose an improved method for error correction.

According to the invention, this object is achieved by a method for error correction of an encoded data stream comprising the steps of:

- saving the demodulated data stream in a small input buffer;
- performing a first correction process on-the-fly in the input buffer;
- transferring the data to an external SDRAM after correction;
- copying a data frame from the external SDRAM to an embedded SRAM;
- starting a multipass correction in the embedded SRAM; and
- copying the corrected data frame back from the embedded SRAM to the external SDRAM.

The method uses a mixture of external SDRAM and internal SRAM. The input/output streaming is performed by a comparatively slow external SDRAM, while the correction is performed in a fast internal SRAM. Therefore, the data from the external SDRAM are copied into the fast internal SRAM only for correction. After the correction process, the so-called "Inner 1", the data is streamed back to the external SDRAM. In this way the number of random accesses to the external SDRAM is reduced. If the errors are sorted, the "Inner 1" correction process and the transfer of the data can be performed at same time. Furthermore, the size of the internal SRAM is also reduced. The internal SRAM correction simplifies the hardware complexity during the correction process.

Favourably, the data stream is encoded with a Reed-Solomon product code. This type of error correction code is widely used for coding data streams on recording media.

A method according to the invention is advantageously performed by a device for error correction of an encoded data stream. Such a device has the advantage that the required SRAM and the random accesses to the SDRAM are reduced. Furthermore, due to

the internal SRAM approach high multipass correction is enabled.

Favourably, a method or a device according to the invention is used by an apparatus for reading from and/or writing to recording media.

For a better understanding of the invention, an exemplary embodiment is specified in the following description with reference to the figures. It is understood that the invention is not limited to this exemplary embodiment and that specified features can also expediently be combined and/or modified without departing from the scope of the present invention. In the figures:

Fig. 1 shows a block diagram of acquisition and the horizontal pass 1;

Fig. 2 shows the process flow of the multipass correction; and

Fig. 3 shows a data flow timing of SRAM2/SDRAM.

In the following the invention is described with reference to optical systems used for digital versatile disks. Of course, the invention is also applicable to other types of disk systems.

On a DVD the input data are packed in ECC blocks. An error correction code (ECC) block comprises 208 rows times 182 columns of symbols, whereby one symbol corresponds to one byte.

For decoding a DVD data frame the following procedures are performed:

- Input and sort the data according to the frame numbers
- Deinterleave the ECC block

- Correct ECC horizontal (PI-correction; inner correction)
- Correct ECC vertical (PO-correction; outer correction)
- Multipass: Perform horizontal and vertical correction again, if necessary
- Descramble data frame
- Perform EDC (error detection code) check on data frame

Two types of corrections are processed in the data flow:

- Horizontal: correcting 208 rows with (182,172,11) RS codes
- Vertical: correcting 182 columns with (208,192,17) RS codes

According to the invention an input buffer 2 and one SRAM bank 11 holding the data for one ECC block is used. The first correction pass, the so-called "inner1" correction, is done on-the-fly, while the multipass correction is done in the embedded SRAM bank 11. A brief description of the data-Flow of the two correction processes is given in the following.

For the first correction process, the demodulated data stream of a set ECC_n is saved in a small embedded SRAM1 2, referred as "input buffer". The correction of the set ECC_n is performed on-the-fly in the embedded SRAM1 2. After correction of inner1 the data of the set ECC_n are transferred to an external SDRAM 3.

For the second correction process, after receiving a full set of ECC_n in the external SDRAM 3 the multipass corrected data of a previous set ECC_{n-1} is read out from the embedded SRAM2 11 back to the external SDRAM 3 while the block ECC_n is copied fro the SDRAM 3 to the SRAM2 11. After receiving a full set of ECC_n in the embedded SRAM2 11, the multipass correction is started. Both processes work independently from each other.

In Fig. 1 a block diagram of acquisition and the horizontal pass 1 is shown. An input controller 1 saves one row of ECC_n in the input buffer 2. The input controller 1 detects streaming discontinuities. Any discontinuities of less than one row

(byte/frame) are corrected immediately in the input buffer 2. If streaming discontinuities of more than one row are detected, this information is stored and the row is written to the correct position of the SDRAM 3 by a deinterleaver 4. After completing one row, the input controller 1 starts a horizontal syndromes unit 5. The horizontal syndromes unit 5 reads the row from the input buffer 2 and computes modified syndromes and an erasure polynomial. A key solver and Chien unit 6 solves the equation and transfers the error values and positions to a correction unit 7. The positions of the first ten acquisition errors are stored in a first erasure memory 8. The correction unit 7 corrects the data of block ECC_n in the input buffer 2 in a bytewise access row by row. If a row is uncorrectable, the row number is marked as uncorrectable. Therefore, a status bit is attached to the row, indicating if the obtained codeword is correct or not. After correction one row, the row is transferred to the external SDRAM 3 via the deinterleaver 4, taking into account the row number information from the input controller 1. If necessary, the deinterleaver 4 jumps row-wise in the SDRAM 3 to correct streaming discontinuities detected by the input controller 1. After acquisition of one complete ECC_n, the second process of multipass correction is initiated.

After receiving one full set of ECC_n in the SDRAM 3, the data are copied to the SRAM2 11. While new ECC is streamed through the SRAM1 2, the vertical and subsequently the multipass correction is performed in the embedded SRAM2 11. The process flow of the multipass correction is shown in Fig. 2. In the figure, in case a pair of numbers is given, the numbers in brackets refer to horizontal correction, while the numbers without brackets refer to vertical correction. A copy unit 9 copies the data of the ECC into the embedded SRAM2 11. It further writes the status bit into the status memory 12 of the "Inner1" pass and computes the erasure positions of uncorrectable errors, which are written into a second erasure memory 10. The second erasure memory 10 stores the positions of

up to 16 rows/10 columns, which were uncorrectable in the previous correction process. After receiving a full ECC, a control unit 14 is started. The control unit 14 reads the last written status of the codeword from the status memory 12. According to this status the process for the codeword is started. The syndrome generator 5 reads the erasure of the las orthogonal process and computes the syndromes of the codeword. After computation of the syndromes, it starts the key solver and Chien 6 search algorithm with the syndromes and the last orthogonal erasure positions. The correction unit 7 corrects the ECC block in the SRAM2 11 in a bytewise access, saving the status of the codeword back into the status memory 12. An output/descrambler 13 descrambles the data stream, performs an EDC check, and copies the data back to a DRAM track buffer area. The output/descrambler 13 may further perform a sector filtering.

The memory needed for the process flow can be summarized as follows:

- Erasure Memory1, 10 bytes per row of SRAM1
- Status Memory, 390 bit = 49 bytes
- Erasure Memory2, 16 bytes
- SRAM1 < 4kbyte
- SRAM2 37856 bytes

Figure 3 shows a more detailed timing structure of the SRAM behaviour. The streams between SRAM and SDRAM reduces the time left for multipass correction. From the figure it can be seen that the two processes described in Fig. 1 and Fig. 2 are decoupled. The stream from the input buffer 2 to the SDRAM 3 is critically temporally linked to the data input, whereas the streams from the SDRAM 3 to the SRAM2 11 and from the SRAM2 11 to the SDRAM 3 can be transmitted discretionarily whenever the bus to the SDRAM 3 is empty. This only depends on the speed of the SDRAM buffer.

Claims

1. Method for error correction of an encoded data stream comprising the steps of:
- saving the demodulated data stream in a small input buffer 2;
- performing a first correction process on-the-fly in the input buffer 2;
- transferring the data to an external SDRAM 3 after correction;
- copying a data frame from the external SDRAM 3 to an embedded SRAM 11;
- starting a multipass correction in the embedded SRAM 11; and
- copying the corrected data frame back from the embedded SRAM 11 to the external SDRAM 3.

2. Method according to claim 1, characterized in that the data stream is encoded with a Reed-Solomon product code.

3. Device for error correction of an encoded data stream, characterized in that it performs a method according to one of claims 1 or 2.

4. Apparatus for reading from and/or writing to recording media, characterized in that it uses a method according to anyone of claims 1 to 2 or comprises a device according to claim 3 for error correction of an encoded data stream.

Abstract

Error correction method for Reed-Solomon product code

The present invention relates to soft-decision decoding of Reed-Solomon product codes.

According to the invention, a method for error correction of an encoded data stream comprises the steps of:

- saving the demodulated data stream in a small input buffer 2;
- performing a first correction process on-the-fly in the input buffer 2;
- transferring the data to an external SDRAM 3 after correction;
- copying a data frame from the external SDRAM 3 to an embedded SRAM 11;
- starting a multipass correction in the embedded SRAM 11; and
- copying the corrected data frame back from the embedded SRAM 11 to the external SDRAM 3.

(Fig. 2)

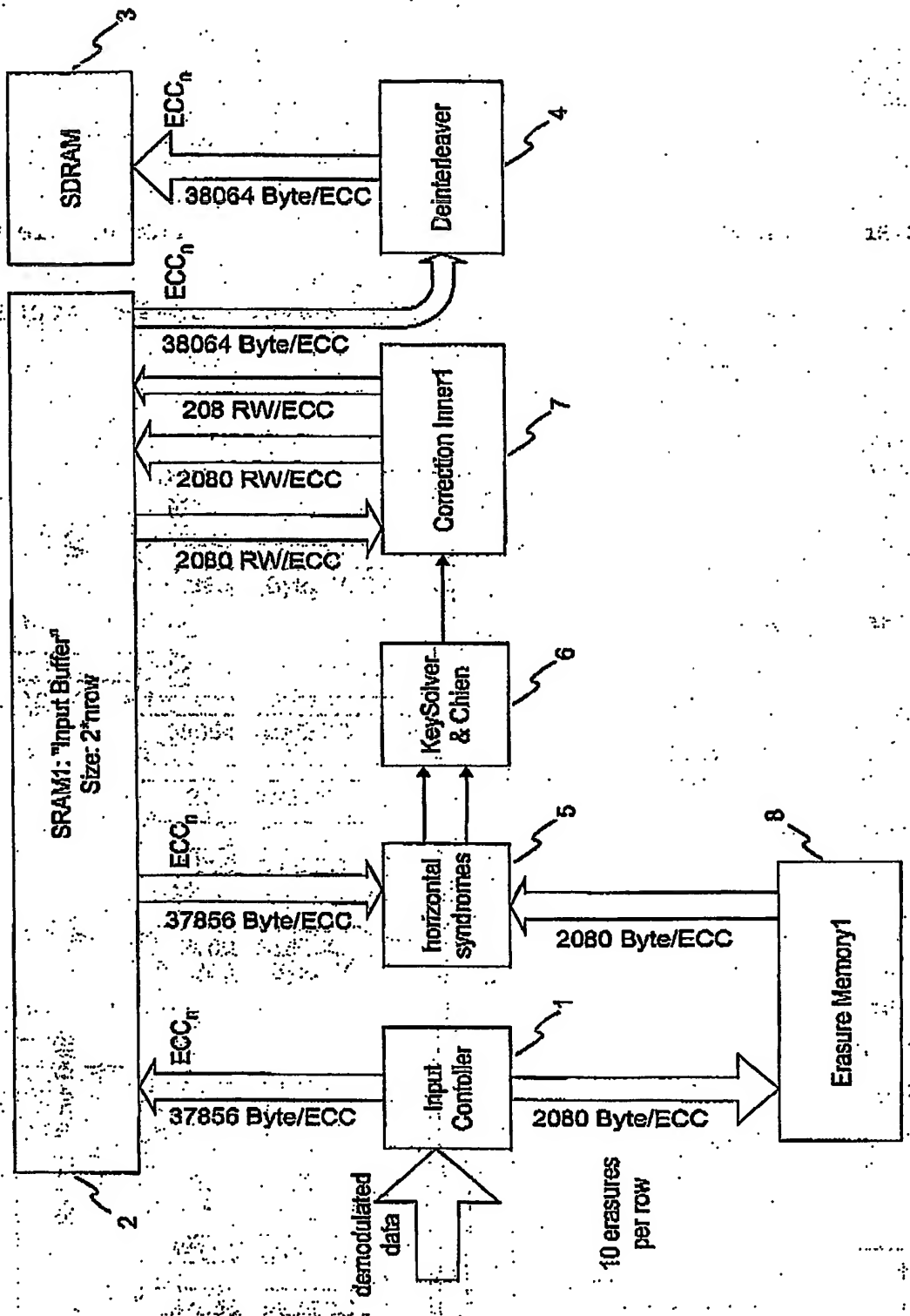
3
SDRAM
ECCn
38064 Byte/ECC
Deinterleaver
4
ECCn
38064 Byte/ECC
208 RW/ECC
2080 RW/ECC
2080 RW/ECC
Correction Inner1
7
SRAM1: "Input Buffer"
Size: 2*nrow
KeySolver & Chien
6
ECCn
37856 Byte/ECC
horizontal syndromes
5
8
2080 Byte/ECC
Erasure Memory1
ECCn
37856 Byte/ECC
Input Controller
1
2080 Byte/ECC
2
demodulated data
10 erasures per row

Fig. 1

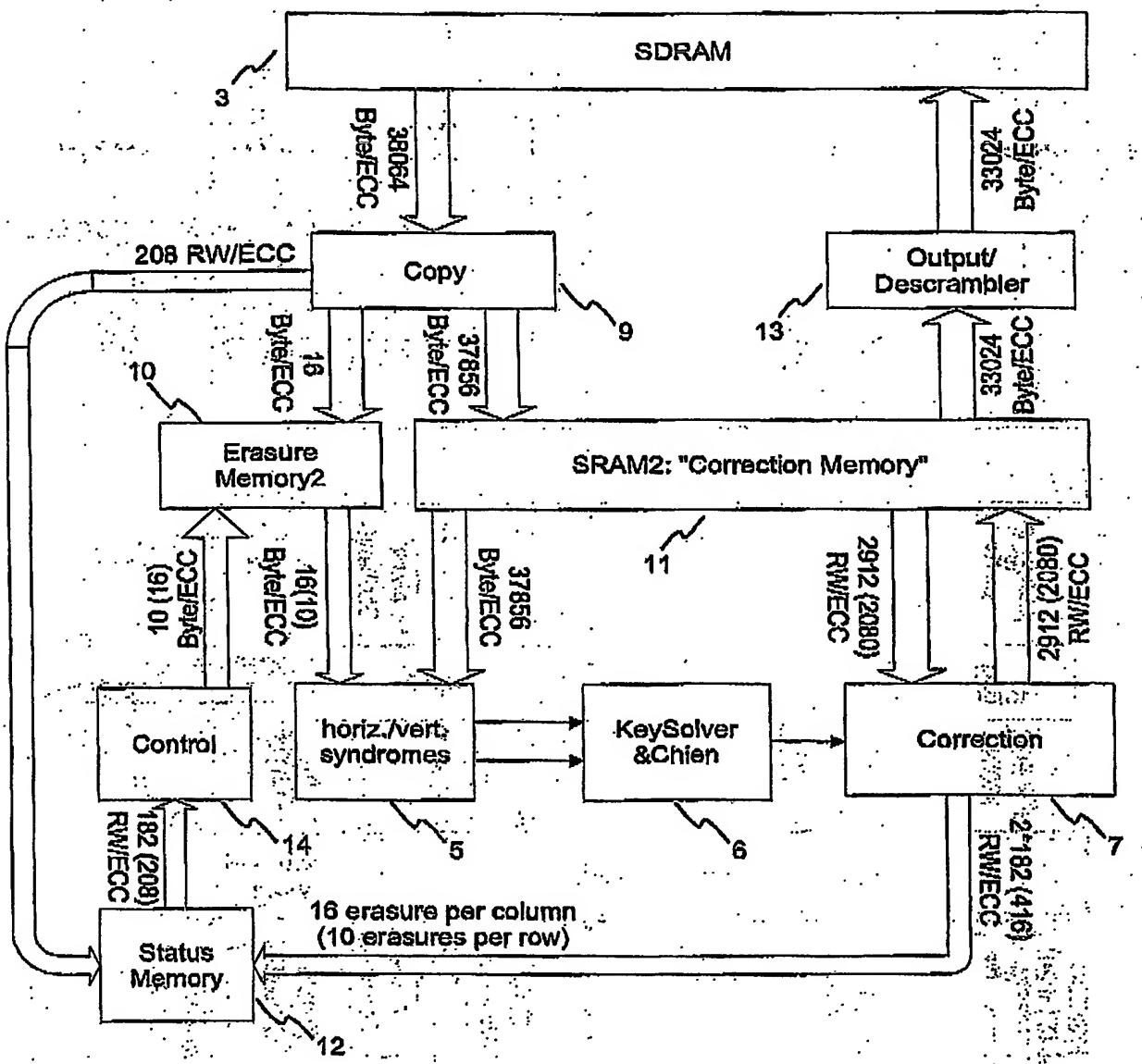
SDRAM
3
38064 Byte/ECC
33024 Byte/ECC
208 RW/ECC
Copy
9
Output/ Descrambler
13
16 Byte/ECC
37856 Byte/ECC
33024 Byte/ECC
10
Erasure Memory2
SRAM2: "Correction Memory"
11
10 (16) Byte/ECC
16(10) Byte/ECC
37856 Byte/ECC
2912 (2080) RW/ECC
2912 (2080) RW/ECC
Control
horiz./vert. syndromes
KeySolver &Chien
Correction
14
5
6
7
182 (208) RW/ECC
2*182 (416) RW/ECC
16 erasure per column
(10 erasures per row)
Status Memory
12

Fig. 2

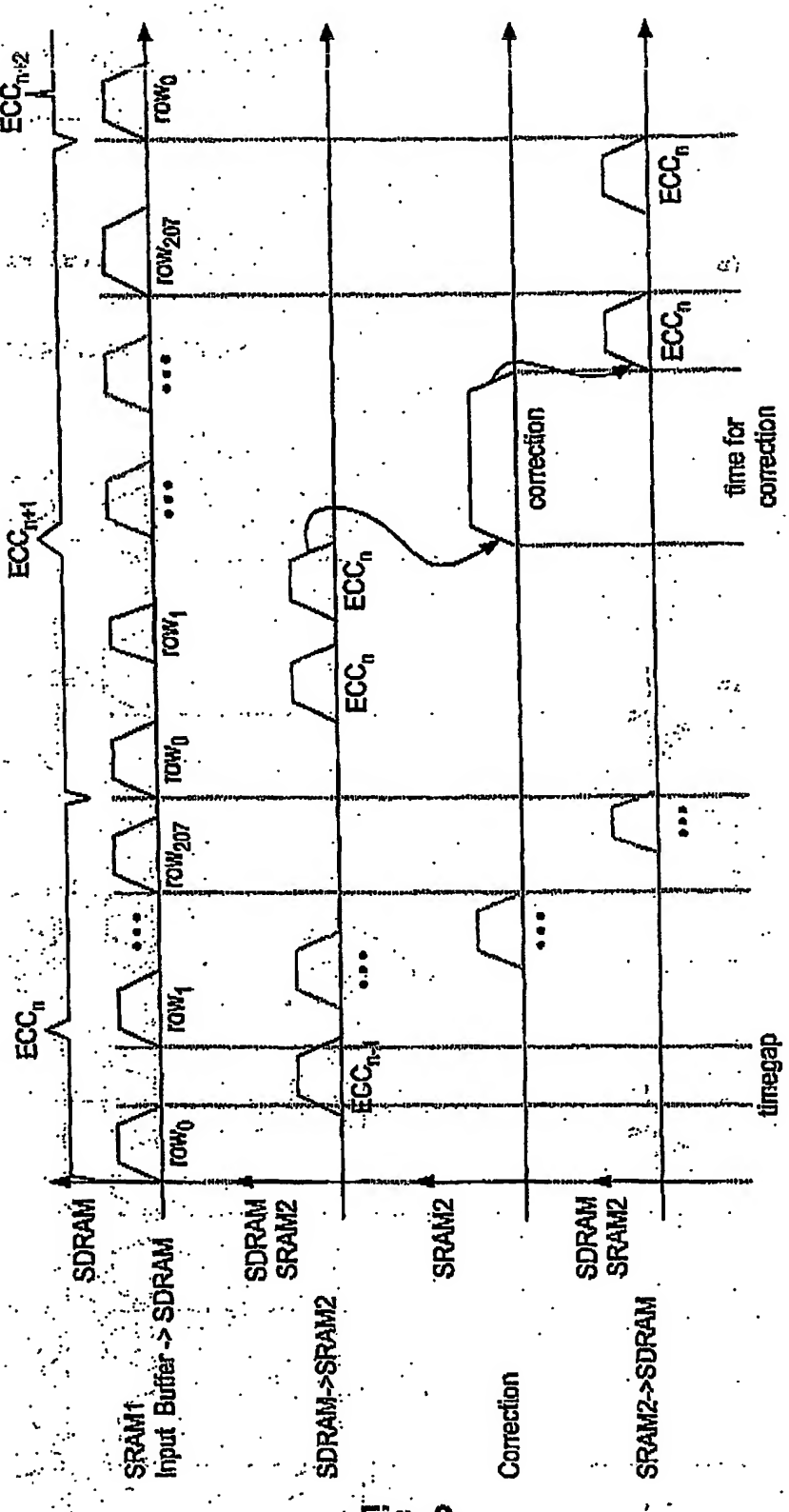
SDRAM
SRAM1
Input Buffer -> SDRAM
SDRAM
SRAM2
SDRAM->SRAM2
SRAM2
Correction
SDRAM
SRAM2
SRAM2->SDRAM
ECCn+2
ECCn+1
ECCn
ECCn-1
row0
row1
row207
correction
time for correction
timegap

Fig. 3